

Amendments to the Claims:

Re-write the claims as set forth below. This listing of claims will replace all prior versions and listings, of claims in the application:

Listing of Claims:

1. (currently amended) A graphics chip comprising:
a front-end in the graphics chip configured to receive one or more graphics instructions and to output a geometry;
a back-end in the graphics chip configured to receive said geometry and to process said geometry into one or more final pixels to be placed in a frame buffer;
wherein said back-end in the graphics chip comprises multiple parallel pipelines.
2. (currently amended) The graphics chip of claim 1 further comprising:
a setup unit for directing said geometry into one of said multiple ~~parallel~~ parallel pipelines.
3. (original) The graphics chip of claim 2 wherein said geometry is determined to locate in a portion of an output screen defined by a tile.
4. (currently amended) The graphics chip of claim 1 wherein each of said parallel pipelines further ~~comprises~~ comprises:
a FIFO unit for load ~~balancing~~ balancing said each of said pipelines.
5. (original) The graphics chip of claim 1 wherein each of said parallel pipelines further comprises:

a scan converter;
a rasterizer;
a unified shader; and
a texture unit.

6. (currently amended) The graphics chip of claim 5 wherein each of said parallel pipelines ~~further~~further comprises:

a z buffer logic unit; and
a color buffer logic unit.

7. (currently amended) The graphics chip of claim 6 wherein said z buffer logic unit interfaces with said scan converter through a ~~hierarchical~~hierarchical Z interface and an early Z interface.

8. (original) The graphics chip of claim 6 wherein said z buffer logic unit interfaces with said unified shader through a late Z interface.

9. (currently amended) A method for processing computer graphics comprising:
receiving one or more graphics instructions in a front-end of a graphics chip and outputting a geometry;
receiving said geometry in a back-end of ~~[[a]]the~~ graphics chip; and
processing said geometry into one or more final pixels to be placed in a frame buffer, wherein said back-end comprises multiple parallel pipelines.

~~[[10]]10.~~ (currently amended) The method of claim 9 further comprising:

using a setup unit to direct said geometry into one of said multiple ~~parallel-parallel~~ pipelines.

11. (original) The method of claim 10 wherein said geometry is determined to locate in a portion of an output screen defined by a tile.

12. (currently amended) The method of claim 9 further comprising:

using a FIFO unit for load ~~balancing-balancing~~ each of said pipelines.

13. (original) The method of claim 9 wherein each of said parallel pipelines further comprises:

a scan converter;
a rasterizer;
a unified shader; and
a texture unit.

14. (original) The method of claim 13 wherein each of said parallel pipelines further comprises:

a z buffer logic unit; and
a color buffer logic unit.

15. (currently amended) The method of claim 14 wherein said z buffer logic unit interfaces with said scan converter through a ~~hierarchical-hierarchical~~ Z interface and an early Z interface.

16. (original) The method of claim 14 wherein said z buffer logic unit interfaces with said unified shader through a late Z interface.

17. (original) A computer program product comprising:
a computer usable medium having computer readable program code embodied therein configured to process computer graphics, said computer program product comprising:
computer readable code configured to cause a computer to receive one or more graphics instructions in a front-end of a graphics chip and output a geometry;
computer readable code configured to cause a computer to receive said geometry in a back-end of a graphics chip; and
computer readable code configured to cause a computer to process said geometry into one or more final pixels to be placed in a frame buffer,
wherein said back-end comprises multiple parallel pipelines.

18. (currently amended) The computer program product of claim 17 further comprises:
computer readable code configured to use a setup unit to direct said geometry into one of said multiple ~~parallel~~ parallel pipelines.

19. (original) The computer program product of claim 18 wherein said geometry is determined to locate in a portion of an output screen defined by a tile.

20. (currently amended) The computer program product of claim 17 wherein said computer readable code configured to cause a computer to process further comprises:

computer readable code configured to cause a computer to use a FIFO unit for load ~~balancing~~balancing each of said pipelines.

21. (original) The computer program product of claim 17 wherein said wherein each of said parallel pipelines further comprises:

a scan converter;
a rasterizer;
a unified shader; and
a texture unit.

22. (currently amended) The computer program product of claim 21 wherein ~~wherein~~ each of said parallel pipelines further comprises:

a z buffer logic unit; and
a color buffer logic unit.

23. (currently amended) The computer program product of claim 22 wherein said z buffer logic unit interfaces with said scan converter through a ~~hierarchical~~hierarchical Z interface and an early Z interface.

24. (original) The computer program product of claim 22 wherein said z buffer logic unit interfaces with said unified shader through a late Z interface.